AN ANALYTIC FORMULATION OF KNOWLEDGE-BASED SYSTEMS FOR INTELLIGENT MACHINES

NAGW-1333

By:

G.N. Saridis

Department of Electrical, Computer and Systems Engineering
Department of Mechanical Engineering, Aeronautical Engineering & Mechanics
Rensselaer Polytechnic Institute
Troy, New York 12180-3590

CIRSSE Document #3

AN ANALYTIC FORMULATION OF KNOWLEDGE-BASED SYSTEMS FOR INTELLIGENT MACHINES

by

George N. Saridis
ECSE Department
Rensselaer Polytechnic Institute
Troy, New York 12180–3590

ABSTRACT

Machines with enough intelligence to perform autonomous tasks in uncertain environments are recently under study. Concepts from the fields of Artificial Intelligence, Operations Research, and Control Theory have been combined to form a unified theory which analytically describes the design and operation of an Intelligent Machine. A summary of the work, aimed to formulate analytically the theory of intelligent machines is presented. The functions of an Intelligent Machine are executed by Intelligent Controls. The Principle of Increasing Precision with Decreasing Intelligence is used to form a hierarchical structure of the control systems. Distributed Intelligence in compatible with such a structure when it is used for teams of intelligent machines or cooperating coordinators within the machine. The three levels of the Intelligent Control, e.g. the Organization, Coordination and Execution Levels are described as originally conceived. New designs as Neural–nets for the organization level and Petri–nets for the coordination level are also proposed. Application to Intelligent Robots for space exploration are suggested.

1. INTRODUCTION

In the last fifteen years, several research efforts have been dedicated to the development of working models for <u>Intelligent Machines</u> as a means to implement human intelligence to machines (Albus 1975, Meystel 1985, Pao 1986, Zames 1979, etc.).

Saridis proposed in (1977) an analytic approach for the design of intelligent machines.

<u>Def. 1</u> **Intelligent Machines are machines that are designed to perform anthropomorphic tasks with minimum interaction with a human operator.**

The function that drives an intelligent machine is called Intelligent Control. Many publications by Saridis and his colleagues (1977, 1979, 1983, 1985a,b,c, 1988a,b) have attempted to generate the component of an analytical design methodology. However, since the theory is not yet complete, there is no comprehensive publication summarizing and integrating the existing results.

This article is an attempt to summarize the up–to–date research efforts and present a complete picture of the theory of intelligent controls as a means of implementation of the intelligent machines.

After a short review of the underlying mathematical theory given in Section 2, the definition of the pertinent variables describing the intelligent machine are given in Section 3, along with the basic principle of Increasing Precision with Decreasing Intelligence. This principle, though originally conceived as a device for simplification of the structure of the machine, has proven to be a generic principle that governs the interaction of machine intelligence with the complexity of its operation. Section 4 discusses the compatibility of hierarchical structures with the concept of distributed intelligence. Sections 5,6, and 7 describe the analytic function of the three levels of an intelligent machine while sections 8 and 9 propose application to robotic systems and give the necessary conclusions.

2. THE MATHEMATICAL THEORY OF INTELLIGENT CONTROLS

Intelligent Machines require control functions in order to perform intelligent functions such as simultaneous utilization of a memory, learning, or multilevel decision making in response to "fuzzy" or qualitative commands. Intelligent Controls have been developed by Saridis (1977, 1983) to implement such functions. They utilize the results of cognitive systems research effectively with various mathematical programming control techniques (Birk & Kelley, 1981).

Cognitive systems have been traditionally developed as part of the field of artificial intelligence to implement, on a computer, functions similar to one encountered in human behavior (Albus 1975, Minsky 1972, Winston 1977, Nilsson 1969, Pao 1986). Such functions as speech recognition and analysis, image and scene analysis, data base organization and dissemination, learning and high–level decision making, have been based on methodologies emanating from a simple logic operation to advances reasoning as in pattern recognition, linguistic and fuzzy set theory approaches. The results have been well documented in the literature.

Various pattern recognition, linguistic or even heuristic methods have been used to analyze and classify speech, images or other information coming in through sensory devices as part of the cognitive system (Birk & Kelley 1981). Decision making and motion control were performed by a dedicated digital computer using either kinematic methods, like trajectory tracking, or dynamic methods based on compliance, dynamic programming or even approximately optimal control (Saridis and Lee 1979).

The theory of Intelligent Control systems, proposed by Saridis (1979) combines the powerful high–level decision making of the digital computer with advanced mathematical modeling and synthesis techniques of system theory with linguistic methods of dealing with imprecise or incomplete information. This produces a unified approach suitable for the engineering needs of the future. The theory may be thought of as the result of the intersection of the three major disciplines of Artificial Intelligence, Operations Research and Control Theory (Figure 1). This research is aimed to establish Intelligent Controls as an engineering discipline, and it plays a central role in the design of Intelligent Autonomous Systems.

Intelligent control can be considered as a fusion between the mathematical and linguistical methods and algorithms applied to systems and processes. They utilize the results of cognitive systems research effectively with various mathematical programming control techniques.

The control intelligence is hierarchically distributed according to the Principle of Increasing Precision with Decreasing Intelligence (IPDI), evident in all hierarchical management systems. They are composed of three basic levels of controls even though each level may contain more than one layer of tree–structured functions (Figure 2):

1. The organization level.
2. The coordination level.
3. The execution level.

The Organization Level is intended to perform such operations as planning and high level decision making from long term memories. It may require high level information processing such as the knowledge based systems encountered in Artificial Intelligence. These require large quantities of knowledge processing but require little or no precision.

The functions involved in the upper levels of an intelligent machine are imitating functions of human behavior and may be treated as elements of knowledge–based systems. Actually, the activities of planning, decision making, learning, data storage and retrieval, task coordination, etc. may be thought of as knowledge handling and management. Therefore, the flow of knowledge in an intelligent machine may be considered as the key variable of such a system.

Knowledge flow in an intelligent machine's organization level represents respectively (Figure 3):

1. Data Handling and Management.
2. Planning and Decision performed by the central processing units.
3. Sensing and Data Acquisition obtained through peripheral devices.
4. Formal Languages which define the software.

Subjective probabilistic models or fuzzy sets are assigned to the individual functions. Thus, their entropies may be evaluated for every task executed. This provides an analytical measure of the total activity.

Artificial Intelligence methods also applicable for the processing of knowledge and knowledge rates of the organization level of an intelligent machine have been developed by Meystel (1985) and his colleagues. Neural-nets have been recently explored as a possible method to implement the organization level (Saridis, Moed 1988).

The Coordination Level is an intermediate structure serving as an interface between the organization and execution level.

It is involved with coordination, decision making and learning on a short term memory, e.g., a buffer. It may utilize linguistic decision schemata with learning capabilities defined in Saridis and Graham (1984), and assign subjective probabilities for each action. The respective entropies may be obtained directly from these subjective probabilities. Recently Petri-nets are investigated for the same reason (Wang, Saridis 1988).

The Execution Level executes the appropriate control functions. Its performance measure can also be expressed as an entropy, thus unifying the functions of an "intelligent machine".

Optimal control theory utilizes a non-negative functional of the states of a system in the states space, and a specific control from the set of all admissible controls,to define the performance measure for some initial conditions, representing a generalized energy function. Minimization of the energy functional yields the desired control law for the system.

For an appropriate density function $p(x, u(x,t), t)$ satisfying Jaynes' Maximum entropy principle (1957), the entropy $H(u)$ for a particular control action $u(x,t)$, is equivalent to the expected energy or cost functional of the system (Saridis 1984). Therefore, minimization of the entropy $H(u)$ yields the optimal control law of the systems.

This statement establishes equivalent measures between information theoretic and optimal control problems and unifies both information and feedback control theories with a common measure of performance. Entropy satisfies the additive property, and any system composed of a combination of such subsystems can be optimized by minimizing its total entropy. Information theoretic methods based on entropy may apply (Conant 1976).

Since all levels of a hierarchical intelligent control can be measured by entropies and their rates, then the optimal operation of an "intelligent machine" can be obtained through the solution of mathematical programming problems.

An important development of this theory is a structure of the "nested hierarchical" systems (Meystel, 1986). Even when the hierarchy is not tree-like, still using hierarchy is beneficial since the hierarchy of resolutions (errors per level) helps to increase the effectiveness of the system under limited computing power which is important to mobile systems.

The various aspects of the theory of hierarchically intelligent controls may be summarized as follows:

The theory of intelligent machines may be postulated as the mathematical problem of finding the right sequence of decisions and controls for a system structured according to the principle of increasing precision with decreasing intelligence (constraint) such that it minimizes its total entropy.

The above analytic formulation of the "intelligent machine problem" as a hierarchically intelligent control problem is based on the use of entropy as a measure of performance at all the levels of the hierarchy. It has many advantages because of the tree-like structure of the decision making process, and brings together functions that belong to a variety of disciplines. The complete development of this theory and its integration with the other theoretical issues of the Intelligent Autonomous System is the main task of this paper.

3. SOME DEFINITIONS AND THE IPDI

3.1 Definitions

It remains to investigate the general concepts of Intelligent Control Systems which pertain to the fundamental functions of Intelligent Machines. Such are the notions of Machine Knowledge, its Rate and Precision.

Def. 2 **Machine Knowledge is defined to be the structured information acquired and applied to remove ignorance or uncertainty about a specific task pertaining to the Intelligent Machine.**

Knowledge is a cumulative quantity accrued by the machine and cannot be used as a variable to execute a task. Instead, the Rate of Machine Knowledge is a suitable variable.

Def. 3 **Rate of Machine Knowledge is the flow of knowledge through an Intelligent Machine.**

Intelligence is defined by the American Heritage Dictionary of the English Language (1969) as: Intelligence is the capacity to acquire and apply knowledge.

In terms of Machine Intelligence, this definition may be modified to yield:

Def. 4 **Machine Intelligence (MI) is the variable (source) which operates on a data-base (DB) of events to produce flow of knowledge (RK)**

One may directly apply the Law of Partition of Information Rates of Conant (1976) to analyze the functions of intelligence within the activities of an Intelligent Control System.

On the other hand, one may define Precision as follows:

Def. 5 **Imprecision is the uncertainty of execution of the various tasks of the Intelligent Machine.**

and

Def. 6 Precision is the complement of Imprecision, and represents the complexity of a process.

Analytically, the above relations may be summarized as follows:

Knowledge (K) representing a type of information may be represented as

$$K = -\alpha - lnp(K) \tag{1}$$

where $p(K)$ is the probability density of Knowledge.

From equation (1) the probability density function $p(K)$ satisfies the following expression in agreement with Jaynes' principle of Maximum Entropy (1957):

$$p(K) = e^{-\alpha-K}; \quad \alpha = ln \int_X e^{-K} dx \tag{2}$$

The Rate of Knowledge R which is the main variable of an intelligent machine with discrete states is defined over a fixed interval of time T:

$$R = \frac{K}{T}$$

It was intuitively thought (Saridis 1983), that the Rate of Knowledge must satisfy the following relation which may be thought of expressing the principle of Increasing Precision with Decreasing Intelligence

$$(MI) : (DB) \longrightarrow (R) \tag{3}$$

A special case with obvious interpretation is, when R is fixed, machine intelligence is largest for a smaller data base e.g. complexity of the process. This is in agreement with Vamos' theory of Metalanguages (1986).

It is interesting to notice the resemblance of this entropy formulation of the Intelligent Control Problem with the ε–entropy formulation of the metric theory of complexity originated by Kolomogorov (1956) and applied to system theory by Zames (1979). Both methods imply that an increase in Knowledge (feedback) reduces the amount of entropy (ε–entropy) which measures the uncertainty involved with the system.

An analytic formulation of the above principle derived from simple probabilistic relation among the Rate of Knowledge, Machine Intelligence and the Data Base of Knowledge, is presented in the next section. The entropies of the various functions come naturally into the picture as a measure of their activities.

3.2 THE ANALYTIC FORMULATION OF THE IPDI

In order to formulate mathematically the concepts of knowledge–based systems, one must consider the state space of knowledge Ω_s with states $s_i, i = 1,2,\ldots n$. They represent the state of events at the nodes of a network defining the stages of a task to be executed.

Then knowledge between two states is considered as the association of the state s_i with another state s_j and is expressed as

$$K_{ij} = \frac{1}{2} w_{ij} s_i s_j \tag{4}$$

where w_{ij} are state transition coefficients, which are zero in case of inactive transmission.

Knowledge at the state s_i is the association of that state with all the other active states s_j and is expressed as

$$K_i = \frac{1}{2} \sum_j w_{ij} s_i s_j \tag{5}$$

Finally, the total knowledge of a system is considered as

$$K = \frac{1}{2} \sum_i \sum_j w_{ij} s_i s_j \tag{6}$$

and has the form of energy of the underlying events. The rate (flow) of knowledge is the derivative of knowledge and for the discrete state space Ω_s is defined respectively

$$R_{ij} = \frac{K_{ij}}{T} \quad , \quad R_k = \frac{K_i}{T} \quad , \quad R = \frac{K}{T} \tag{7}$$

where T is a fixed time interval.

Since knowledge was defined as structured information, it can be expressed by a probabilistic relation similar to the one given by Shannon, and expressed for each level by equation (1):

$$ln p(K_i) = -\alpha - K_i \tag{8}$$

which yields a probability distribution satisfying Jaynes' Principle of Maximum Entropy (for $E\{K\} = \text{Const.}$)

$$p(K_i) = e^{-\alpha_1 - K_i} \qquad e^{\alpha_1} = \sum_i e^{-K_i}$$

The rate of knowledge is also related probabilistically by considering that $K_i = R_i T$.

$$p(R_i) = p(R_i T) = e^{-\alpha_1 - TR_i} = e^{-\alpha_1 - \mu_1 R_i} \tag{9}$$

The principle of Increasing Precision with Decreasing Intelligence is expressed probabilistically by

$$PR(MI, DB) = PR(R) \tag{10}$$

where MI is the machine intelligence and DB is the data base associated with the task to be executed and represents the complexity of the task which is also proportional to the precision of execution. The following relation is obtained by conditioning and taking the natural logarithms:

$$lnp(MI/DB) + lnp(DB) = lnp(R) \tag{11}$$

Taking the expected value on both sides

$$H(MI/DB) + H(DB) = H(R) \tag{12}$$

where $H(x)$ is the entropy associated with x. For a constant rate of knowledge which is expected during the conception and execution of a task increase of the entropy of DB requires a decrease of the entropy of MI for the particular data base, which manifests the IPDI. If MI is independent of DB then

$$H(MI) + H(DB) = H(R) \tag{13}$$

In the case that $p(MI)$ and $p(DB)$ satisfy Jaynes' principle as $p(R)$ does, where

$$\begin{aligned} p(MI/DB) &= e^{-\alpha_2-\mu_2 MI_{DB}} \\ p(DB) &= e^{-\alpha_3-\mu_3 DB} \end{aligned} \tag{14}$$

where α_i and $\mu_i, i = 2,3$ are appropriate constants.
Then the entropies are rewritten as

$$-\alpha_2 - \mu_2 MI_{DB} - \alpha_3 - \mu_3 DB = -\alpha_1 - \mu_1 R \tag{15}$$

and if

$$\alpha_1 = \alpha_2 + \alpha_3 \qquad \gamma_2 = \frac{\mu_2}{\mu_1} \ , \ \gamma_3 = \frac{\mu_3}{\mu_1}$$

then

$$\gamma_2 MI_{DB} + \gamma_3 DB = R \tag{16}$$

which represents a specific but more explicit version of the Principle of Increasing Precision with Decreasing Intelligence.

This Principle is applicable both across one level of the Intelligent Hierarchy as well as throughout the levels of the Hierarchy, in which case the flow R represents the throughput of the system in an information theoretic manner. The partition law of information rate applies naturally to such a system.

The entropy of DB may be related to ϵ–entropy as follows: A system requiring certain (n) level of precision takes n–times the data base DB required for a simple precision. But

$$H(nDB) = E\{lnn\} + E\{lnDB\} \tag{17}$$

where $E\{lnn\}$ is the ϵ–entropy associated with the complexity of execution. A case study demonstrating the validity of the above is given in Saridis and Valavanis (1988).

4. DISTRIBUTED MACHINE INTELLIGENT SYSTEMS

In the real world, distributed systems and hierarchical systems co–exist in harmony. The human organism is a typical example of this statement.

Distributed Artificial Intelligence (DAI) is a discipline concerned with treating problems that require multiple solvers in parallel by invoking artificial intelligence techniques (Decker, 1987). When utilized to control intelligent machines working in parallel, it can be interpreted as Distributed Machine Intelligence (DMI) where the intelligence processing is referred to the autonomous abilities of the machines involved as with simple hierarchically intelligent control ease, (Saridis, 1986): This corresponds more to the distributed problem solving process and may be thought of as composed of two components:

<u>Distributed Machine Intelligence</u>

- Control
- Communications.

Distributed Control can be performed in two different ways:

- Control by a meta level
- Control by majority vote.

The first method is an extension of the hierarchical approach where the coordination, decision making and subtask assignment is deferred to a higher level of intelligence imbedded in the dispatcher of the intelligent machines; (see Fig. 4). The cooperative activities should be planned, scheduled and sequenced in this device and communicated to the appropriate machines. Feedback from the environment should be communicated continuously for the evaluation of the team work performed.

The second method deals with cooperative approach of machines operating in the same environment and performing tasks that require scheduling and task assignment. Majority vote may provide the proper planning and sequencing of the various tasks to be performed in unison by all the intelligent machines involved. The majority vote could be taken in a poll place equally accessed by al the machine and communicated back to them in the appropriate sequence.

The communication problem plays a paramount role in distributed machine intelligence. It may be performed by a large communication network in the case of wide spacially distributed machines or by a computer bus when dealing with a tightly built system of devices. The main design considerations of a communication system are:

- The system configuration,
- The protocol, and
- The treatment of uncertainty of information.

The first item deals with the selection of the proper structure of the network. Two types suitable for the appropriate control categories are

- Star Connection
- Ring Connection.

The second item is essential for the most efficient operation of the system and the optimization of the information exchange among the intelligent machines. The computer literature contains many sources of information about protocols as in Lampson, Paul and Siegert (1981).

The third item deals with ability of the communications system to deal with uncertain and incomplete information. The problem of reliability for accurate and precise transmission and reception of information is essential. The classical Shannon's information theory methods are applicable here (Shannon and Weaver, 1963).

Finally, as mentioned earlier, distributed machine intelligence may be applied to coordinate a number of cooperating intelligent machines or to organize a number of coordinators within the same machine. In both cases, such a structure can work in harmony with the hierarchically intelligent control structure of Saridis (1983). The reason is that the hierarchical stratification refers to the intelligence of the machine and the IPDI needs only to be generalized from a vertical to a horizontal deployment. In other words, the IPDI should be assigned to all directions of flow of knowledge to represent all the trade–offs between intelligence and complexity.

5. THE ORGANIZATION LEVEL AND KNOWLEDGE BASED SYSTEM

The function of the organizer, the highest level of the hierarchy of Intelligent Controls, is based on several AI (knowledge based) concepts forming the foundations of Machine Intelligence. These concepts translated into probabilistic models form the functions of representation and reasoning, planning, decision making, long–term memory exchange and learning through feedback to set up a task in response to some outside command (Fig. 3). The probabilistic model generated provides the mechanism to select the appropriate task for the appropriate command. The principle followed here is that instead of task decomposition a collection of tasks is generated from a list of primitive stored in the memory and matched against the input command applied.

The organization level algorithm must perform the following functions:

- Receive a command and reason about it. Reasoning and representation associates different primitive activities and rules with the received command and evaluates probabilistically each activity.
- Planning which involves operations on the activities. The ordering of the activities and insertion of repetitive primitive events to complete a plan is accomplished according to the selected rules. Transition–matrices (masks) and transition probabilities are used to order the activities and calculate their total probability.
- Decision Making which selects the most probable plan.
- Feedback which updates the probabilities through learning algorithms after the completion of the job, after the completion and evaluation of each task.
- Memory exchanges which updates the stored information in the long–term memory.

To specify analytically the functions of the organizer, it is essential to derive the domain of the operation of the machine for a particular class of problems (Valavanis 1985). Assuming that the environment is known, one may define the following sets:

The set of commands $C = \{c_1, c_2, \ldots, c_m\}$ in natural language, received by the machine as inputs. Each command is compiled to yield an equivalent machine code explained in the next section.

The task command of the machine which contains a number n of independent events.

The events $E = \{e_1, e_2, \ldots, e_n\}$ are individual primitive rules or activities e_i stored in the long–term memory and representing tasks to be executed. The task domain indicates the capabilities of the machine.

Activities A, are groups of events concatenated to define a complex task: e.g., $A_{234} = \{e_2, e_3, e_4\}$. If the events are ordered, then we have an ordered activity.

A random variable $x_i \epsilon [0, 1]$ is associated with each individual event e_i. If the random variable x_i is binary (either 0 or 1), it indicates whether an event e_i is inactive or active, in a particular activity and for a particular command. If the random variables x_i are continuous (or discrete but not binary) over [0,1], they reflect a membership function in a fuzzy decision making problem. At this point, we consider the x_i's to be binary.

Functions F, are internal operations on the activities A. As such, they are defined in their right order within the organization level.

a) Machine Representation and Reasoning, R, is association of the compiled command to a number of activities and/or rules. A probability function is assigned to each activity and/or rule and the Entropy associated with it is calculated. When rules are included one has active reasoning (inference engine).

b) Machine Planning, P, is ordering of the activities. The ordering is obtained through a sparse matrix M of 0's and 1's, which indicate the proper order of the primitive events.

c) Decision Making, DM, is the function of selecting the sequence with the largest probability of success.

d) Feedback, FB, is evaluation of the cost functions and updating of the probabilities associated with each primitive event and activity.

e) Memory Exchange, ME, is retrieval and storage of information from the long-term memory based on selected feedback data from the lower levels after the completion of the complex task.

An algorithm of the functions of the organizer is given below. The received command is related to a random word through reasoning that associates the various strings of events in binary code with appropriate probabilities. Planning and decision making follow, while feedback provides an off-line upgrading of the probabilities through learning algorithms. Long-term memory exchange updates the stored information and related probabilities, and provides the actual job for the coordinators.

The algorithm, which performs a number of sequential functions, is outlined by specifying the following:

1. The set of user commands $C = \{c_1, c_2, \ldots, c_M; M$ fixed and finite$\}$ with associated probability distribution functions (pdfs) $p(c_n), n = 1, 2, \ldots, M$, sent to the Intelligent Machine via some channel.
2. The set of classified compiled input commands $U = \{u_1, u_2, \ldots, u_M$ fixed and finite$\}$ with associated pdfs $p(u_j/c_n), j = 1, 2, \ldots, M$, which are the inputs to the organization level of Intelligent Machines.
3. The task domain of the Intelligent Machine with the set of independent but not mutually exclusive disjoint sub-sets of non-repetitive and repetitive primitive events $E = \{e_{nr}, E_r\} = \{e_1, e_2, \ldots, e_{N-L}, e_{N-1+1}, \ldots e_N; N$ fixed and finite$\}$.
4. The binary valued random variable x_i associated with each e_i indicating if e_i is active $(x_i = 1)$ or inactive $x_i = 0)$ given a u_j, with corresponding pdfs $p(x_i = 1/u_j)$ and $p(x_i = 0/u_j)$ respectively.
5. The set of the $(2^N - 1)$ activities which are groups of primitive events concatenated together to define a complex task. They are represented by a string of binary random variables $X_{jm} = (x_1, x_2, \ldots, x_n)_m, m = 1, 2, \ldots, (2^N - 1)$, which indicates which e_i's are active or inactive within an activity with a pdf $P(X_{jm}/u_j)$.
6. The set of compatible ordered activities obtained by ordering the primitive events within each activity and represented by a string of compatible ordered binary random variable Y_{jmr}, where r denotes the rth ordered activity obtained from X_{jm}, with a

pdf $P(Y_{jmr}/u_j)$.

7. The set of compatible augmented ordered activities obtained by inserting repetitive primitive events within appropriate positions of each Y_{jmr} and represented by $Y_{jmr}(a_s)$. where a_s denotes the sth augmented activity obtained from Y_{jmr} and a pdf $P(Y_{jmr}(a_s)/Y_{jmr})$.
8. The set of mask matrices M_{jmr} with associated pdfs $p(M_{jmr}/u_j)$ used to obtain the compatible ordered activities (Y_{jmr}) from the activities (X_{jm}).
9. The set of augmented mask matrices $M_{jmr}(a_s)$ with associated pdfs $p(M_{jmr}(a_s)/Y_{jmr})$ used to obtain the compatible pdfs $p(M_{jmr}(a_s)/Y_{jmr})$ used to obtain the compatible augmented ordered activities from each Y_{jmr}.
10. The set of rules for the compatibility and completeness test.
11. The learning mechanism for decision making where the Entropies corresponding to the total probabilities are compared for minimum value.
12. The Feedback mechanism which updates the probabilities by learning. through an evaluation of the task execution from the lower levels.

When a user command c_n with a pdf $p(c_n)$ is sent to the Intelligent Machine. it is received and classified by the classifier to yield the (classified) compiled input command u_j with a pdf $p(u_j/c_n)$. which is the input to the organization level.

The organization level formulates complete and compatible plans and decides about the best possible plan to execute the user requested job. This is done by associating u_j with a set of pertinent activities X_{jm} with corresponding probabilities $P(X_{jm}/u_j)$ (reasoning), and by organizing the activities in such a way (planning) to yield complete and compatible plans: The compatible ordered activities Y_{jmr} are obtained via the mast matrices M_{jmr} and their associated pdfs are: $P(Y_{jmr}/u_j) = p(M_{jmr}/u_j)P(X_{jm}/u_j)$. The compatible augmented ordered activities $Y_{jmr}(a_s)$ are obtained by inserting repetitive primitive events in appropriate positions within each Y_{jmr} and their corresponding pdfs are: $P(Y_{jmr}(a_s)/Y_{jmr}) = p(M_{jmr}(a_s)/Y_{jmr})$. $P(Y_{jmr}/u_j)$. Every incompatible activity and incomplete plan is rejected. The most probable complete and compatible plan Y^F is the final plan that is transferred to the coordination level. (see Figure 4).

Each function has been described in a set theoretic manner and probabilities are assigned as measures. Entropies $H(F(X))$ as associated with each function in a straight forward way. Transmissions of information $T(x_i : X_j)$ measure the interdependence between different functions.

The Entropy function is used to calculate the uncertainty of the activities and ordered activities.

Assume that there are S different states of the organizer and that the inputs to the organizer belong to C. It has been shown that the functions of the organization level obey a generalized law of partition of information rates (Conant 1976). According to this law the total activity rate of the organizer is decomposed into the Throughput Rate (flow of Knowledge), the Blockage Rate (Decision Making), Coordination Rate (Planning), Internal Decision Rate (Reasoning and Noise Rate:

$$F = F_T(C:S) + F_B(C:S) + F_C(C:S + F_D(C:S) + F_N(C:S) \qquad (18)$$

where:

$F =$ the total activity rate

$F_T =$ the throughput rate corresponding to information transfer within the organizer

$F_B =$ the blockage rate corresponding to Decision Making

$F_C =$ the coordination rate corresponding to Planning

$F_D =$ the internal decision making rate corresponding to Reasoning

$F_N =$ the noise rate corresponding to information when the command C has been already received.

Learning in the organizer, as well as the entire Intelligent Machine, is obtained through selective feedback from the lower levels. Feedback in the organization level is applied after the completion of a whole task, in contrast to real-time feedback provided to the lower levels. The task is evaluated by cost functions J_c and all the probabilities associated with the organizer are upgraded by the stochastic approximation algorithm:

$$p(t+1) = p(t) + \gamma_{t+1}[\xi - p(t)]; \; \gamma_{t+1} = \frac{1}{t+1}; \; \xi = \begin{cases} 1 & ; \; J = \min J_c \\ 0 & ; \; \text{otherwise} \end{cases} \qquad (19)$$

Convergence of this algorithm has been proven elsewhere (Saridis and Graham 1984), establishing the learning property of the organizer.

A total Entropy is calculated for each final complete plan. This Entropy includes both the reasoning and planning uncertainty. The complete ordered activity with the minimum total Entropy is considered the most likely to execute the job, and is communicated to the coordination level. (Saridis and Valavanis 1988).

Current research has established the potential of using neural nets to perform the functions of the organizer. A rigorous derivation of the Boltzmann machine and its use to connect the nodes of the events in the organization level was presented in Saridis and Moed (1988) and Hinton, Sejnowski (1986).

6. THE COORDINATION LEVEL

The coordination level is an intermediate structure serving as an interface between the organization and the execution level. It is essential for dispatching organizational information to the execution level. Its objective is the actual formulation of the control problem associated with the most probable complete and compatible plan formulated by the organization level that will execute in real–time the requested job.

This includes selection of one among alternative plan scripts that accomplish the same job in different ways according to the constraints imposed by the workspace model and timing requirements, controlled by the dispatcher (Figure 4).

The coordination level is composed of a specified number of coordinators. Specific hardware, (execution devices) from the execution level, is associated with each coordinator. These execution devices execute well defined tasks when a command is issued to them by their corresponding coordinator. (Valavanis 1986). The dispatcher serves as both the communicator of information from the organization level to the coordinators and on–line exchange of data among the coordinators. A Petri–net formulation of these activities has been recently proposed by Wang and Saridis (1988).

The major advantage which results from this association is that the individual functions of each coordinator may be defined a priori (during the design phase of the Intelligent Machines) because they are considered to be unmodifiable with time. Thus, they are assumed to be deterministic functions because the number of parameters involved in each one of them is also pre–specified.

This structure implies that the coordination level does not have any reasoning capabilities like the organizer. Its intelligence is related to its ability on how to execute the organizer plan in the best possible way. The coordination level involves decision making associated with specific knowledge (information) processing based on the already formulated plan utilizing Petri–nets (Wang, Saridis 1988)(Peterson 1977).

The functions of the coordination level are defined in terms of the individual functions of the different coordinators of an Intelligent Machine, i.e. for an Intelligent Robotic System: (see Figure 4)

1. The Vision System Coordinators (VSC).
2. The Sensor System Coordinator (SSC).
3. The Motion Coordinator(s) (MSC), and,
4. The Gripper(s) Coordinator(s) (GSC).

It is important to clarify at this point that we consider the VSC as a separate coordinator and not as a part of the SSC. The main reasons for this distinction are: First, Robotic Vision has become a very important component in modern robotic systems and robotic vision system are studied and treated separately from all other types of external (and internal) sensors. Second, the hardware associated with the VSC is different than the one associated with the SSC, and third, this paper is mainly concerned with the VSC and ignores the details of operation of the other sensory systems.

Each coordinator, when accessed by the dispatcher performs a pre–specified number of different functions. A cost is assigned to each individual function. An accrued cost is associated with the operation of each coordinator. An overall accrued cost is calculated in terms of the weighted sum of the accrued costs of the coordinators after the execution of the requested job. This cost is communicated to the organizer after the completion of the requested job and is used to upgrade the information stored in the long-term memory of the organization level. This feedback information (which is sent from the coordination to the organization level after the completion of the requested job) will be called off-line feedback information learning. On the other hand, feedback information is communicated to the coordination level from the execution level during the execution of the requested job. Each coordinator, when accessed, issues a number of commands to its associated execution devices (at the execution level). Upon completion of the issued commands feedback information is received by the coordinator and is stored in the short-term memory of the coordination level. This information is used by other coordinators if necessary, and also to calculate the individual, accrued and overall accrued costs related to the coordination level. Therefore, the feedback information from the execution to the coordination level will be called on-line, real-time feedback information. More details about the feedback mechanism are given in the corresponding sections where the functions of each coordinator are explained.

7. THE EXECUTION LEVEL WITH ENTROPY FORMULATION

The cost of control problem at the hardware level can be expressed as an entropy which measures the uncertainty of selecting an appropriate control to execute a task. By selecting an optimal control, one minimizes the entropy, e.g., the uncertainty of execution. The entropy may be viewed in the respect as an energy in the original sense of Boltzmann, as in Saridis (1985).

Optimal control theory utilizes a non-negative functional of the states of the system $x(t)\epsilon\Omega_x$ the state space, and a specific control $u(x,t)\epsilon\Omega_u x\Pi, \Omega_u \subset \Omega_x$ the set of all admissible feedback controls, to define the performance measure for some initial conditions $(x_0(t_0)$, representing a generalized energy function, of the form,

$$V(x_0,t_0) = \int_{t_0}^{t_f} L(x,t,u(x,t))dt \tag{20}$$

where $L(x,t,u(x,t)) > 0$, subject to differential constraints dictated by the underlying process

$$\dot{x} = f(x,u(x,t)),t;\ x(t_0) = x_0;\ x(t_f)\epsilon M_f \tag{21}$$

with M_f a manifold in Ω_x. The trajectories of the system (21) are defined for a fixed but arbitrarily selected control $u(x,t)$ from the set of admissible feedback controls Ω_u.

In order to express the control problem in terms of an entropy function one may assume that the performance measure $V(x_0, t_0, u(x,t))$ is distributed in Ω_u according to the probability density $p(u(x,t))$ of the controls $u(x,t)\epsilon\Omega_u$. The entropy $H(u)$ corresponding to this density is defined as

$$H(u) = -\int_{\Omega_u} p(u(x,t)) ln p(u(x,t))dx$$

and represents the uncertainty of selecting a control $u(x,t)$ from all the possible admissible feedback controls from Ω_u. The optimal performance should correspond to the maximum value of the associated density $p(u(x,t))$. Equivalently, the optimal control $u^*(x,t)$ should minimize the entropy function $H(u)$.

This is satisfied if the density function is selected to satisfy Jaynes' Principle of Maximum Entropy (1956), e.g.

$$p(u(x,t)) = c\exp\{= V(x_0, t_0, u(x,t))\} \tag{22}$$

It was shown by Saridis (1985b) that the expression $H(u)$ representing the entropy for a particular control action $u(x,t)$ is given by

$$\begin{aligned} H(u) &= \int_{\Omega_u} p(x, u(x,t))\ V(x,t,u(x,t))dx \\ & E_x\{v(x_0, t_0, u(x,t))\} \end{aligned} \tag{23}$$

This implies that the average performance measure of a feedback control problem corresponding to a specifically selected control, is an entropy function. The optimal control $u^*(x,t)$ that minimizes $V(x,t,u(x,t))$, maximizes $p(x,u(x,t))$, and consequently minimizes the entropy $H(u)$.

$$\begin{aligned} u^* &: E_x\{V(x,t,u^*(x,t))\} \\ &= \min_u \int_{\Omega_u} V(x,t,u(x,t))p(u(x,t))dx \end{aligned} \tag{24}$$

This statement establishes equivalent measures between information theoretic and optimal control problems and provides the information and feedback control theories with a common measure of performance.

8. APPLICATION TO ROBOTIC SYSTEMS

The theory of Intelligent Controls has direct application to the design of Intelligent Robots. The IPDI provides a means of structuring hierarchically the levels of the machine. Since for a passive task the flow of knowledge through the machine must be constant, it assigns the highest level with the highest machine intelligence and smallest complexity (size of data base). and the lowest level with the lowest machine intelligence and largest complexity. Such a structure agrees with the concept of most organizational structures encountered in human societies. Application to machine structures is straight forward.

Even at the present time there is a large variety of applications for intelligent machines. Automated material handling and assembly in an automated factory. automated inspection. sentries in a nuclear containment are some of the areas where intelligent machines have and will find a great use. One of the most important applications though is the unmanned space exploration where. because of the distances involved. autonomous anthropomorphic tasks must be executed and only general commands and reports of executions may be communicated.

Such tasks are suitable for intelligent robots capable of executing anthropomorphic tasks in unstructured uncertain environments. They are structured uncertain environments. They are structured usually in a human– like shape and are equipped with vision and other tactile sensors to sense the environment. two areas to execute tasks and locomotion for appropriate mobility in the unstructured environment. The controls of such a machine are performed according to the theory of Intelligent Machines previously discussed. (Saridis and Stephanou. 1977). (Saridis 1983. 1985a. 1985b. 1988a). (Meystel 1985.1986). The three levels of controls. obeying the Principle of Increasing Precision with Decreasing Intelligence. are implemented with appropriately selected feedback. as shown in Figure 5. for a PUMA 600 robot arm with sensory feedback.

The Boltzmann machine architecture (Saridis and Moed 1988) may be used to implement the organization level of an intelligent robot by considering the proper interconnection of primitive events represented by nodes and the coordination level of an intelligent robot by appropriately connecting the various coordinators to the dispatcher for communications purposes.

9. CONCLUSIONS

A mathematical theory for intelligent machines was proposed and traced back to its origins. The methodology was developed to formulate the "intelligent machine". of which an intelligent robot system is a typical example. as a mathematical programming problem as using the aggregated entropy of the system as its performance measure. The levels of the machine structured according to the Principle of Increasing Precision with Decreasing Intelligence can adopt performance measures easily expressed as entropies. This work establishes an analytic formulation of the Principle. provides entropy measures for the account of the underlying activities. and integrates it with the main theory of "Intelligent Machines". Optimal solutions of the problem of the "intelligent machine" can be obtained by minimizing the overall entropy of the system. The entropy formulation presents a tree–like structure for this decision problem very appealing for real–time computational solutions.

ACKNOWLEDGEMENTS

This work was supported by NSF Grant ENG DMC–8717969.

REFERENCES

Albus, J. S., (1975). "A New Approach to Manipulation Control: The Cerebellar Model Articulation Controller". *Transactions of ASME, J. Dynamics Systems, Measurement and Control, 97,* 220-227.

Athans, M. (1987). "Command and Control (c2) Theory: A Challenge to Control Science", *IEEE Trans. on AC, 32,* No. 4, 286–293, April.

Birk, J. R. and Kelley, R. B., (1981, "An Overview of the Basic Research Needed to Advance the State of Knowledge in Robotics", *IEEE Trans. on SMC,* SMC–11, No. 8, pp. 575–579.

Conant, R. C., (1976). "Laws of Information Which Govern Systems". IEEE Trans. on SMC, SMC–6, 4, 240–255, April 1976.

Decker, K.S. (1987). "Distributed Problem Solving Techniques, A Survey", *IEEE Trans. on Syst., Man and Cybernetics, SMC–17,* No. 5, 729–40, Sept.–Oct.

Fu, K. S., (1971), "Learning Control Systems and Intelligent Control Systems: An Intersection of Artificial Intelligence and Automatic Control", *IEEE Trans on Automatic Control,* Vol. AC–16, No. 1, 70–72.

Hayes–Roth, et al., (1983), *Building Expert Systems,* Addison–Wesley, New York.

Hinton, G. E., Sejnowski, T. J. (1986), "Learning and Relearning in Boltzmann Machines", pp. 282–317, in *Parallel Distributed Processing,* ed, D. E. Rumelhart and J. L. McClellan, MIT Press.

Jaynes, E. T., (1957), "Information Theory and Statistical Mechanics". *Physical Review,* 106, 4.

Kolmogorov, A. N., (1956), "On Some Asymptotic Characteristics of Completely Bounded Metric Systems", Dokl Akad Nank, SSSR, Vol. 108, No. 3, pp. 385–9.

Lampson, B. W., Paul, M., Siegert, H., Eds. (1981), *Distributed Systems Architecture and Implementation,* Springer Verlag, New York.

Meystel, A., (1985), "Intelligent Motion Control in Anthropomorphic Machines", Chapter in *Applied Artificial Intelligence,* S. Andriole Ed. Pentrocellis Books, Princeton, NJ.

Meystel, A., (1986), "Cognitive Controller for Autonomous Systems", *IEEE Workshop on Intelligent Control 1985,* p. 222, RPI, Troy, New York.

Minsky, M. L., (1972), *Artificial Intelligence,* McGraw–Hill, NY.

Nilsson, N. J., (1969). "A Mobile Automation: An Application of Artificial Intelligence Techniques", *Proc. Int. Joint Conf. on AI.* Washington, D. C.

Pao, Y.–H., (1986). "Some Views on Analytic and Artificial Intelligence Approaches", *IEEE Workshop on Intelligent Control,* p. 29, RPI, Troy, NY.

Peterson, J.L., (1977). "Petri Nets", *Computing Surveys", 9.* No. 3, 223–252, Sept.

Sage, (1987), "Information Systems Engineering for Distributed Decision Making".

Saridis, G. N., (1977). *Self-organizing Controls of Stochastic Systems.* Marcel Dekker, New York, New York.

Saridis, G. N., (1979). "Toward the Realization of Intelligent Controls", *IEEE Proceedings,* Vol. 67, No. 8.

Saridis, G. N., (1983). "Intelligent Robotic Control", *IEEE Trans. on AC-29,* 4.

Saridis, G. N., (1985a). "Intelligent Control–Operating Systems in Uncertain Environments", Chapter 7 in *Uncertainty and Control,* J. Ackermann Editor, Springer–Verlag, Berlin, pp. 215–233.

Saridis, G. N., (1985b). "Control Performance as an Entropy", *Control Theory and Advanced Technology,* 1, 2.

Saridis, G. N., (1985c). "Foundations of Intelligent Controls". *Proceedings IEEE Workshop on Intelligent Controls,* p. 23, RPI, Troy, NY.

Saridis, G. N., (1988a). "Entropy Formulation for Optimal and Adaptive Control", *IEEE Transactions on AC,* Vol. 33, No. 8, pp. 713–721.

Saridis, G. N., (1988b), Proceedings 1988 IFAC Conference on Robot Control, Karlsruhe, West Germany, SYROCO '88, Oct.

Saridis, G. N. and Graham, J. H., (1984). "Linguistic Decision Schemata for Intelligent Robots", *Automatica,* Vol. 20, No. 1, 121–126.

Saridis, G. N. and Lee, C. S. G., (1979). "Approximation of Optimal Control for Trainable Manipulators", *IEEE Trans. on SMC,* Vol. SMC–8, No. 3.

Saridis, G. N. and Moed, M. C., (1988). "Analytic Formulation of Intelligent Machines as Neural Nets", Symposium on Intelligent Control, Washington, D.C., August.

Saridis, G. N., Stephanou, H. E., (1977). "A Hierarchical Approach to the Control of a Prosthetic Arm", *IEEE Trans. on SMC,* Vol. SMC–7, No. 6, pp. 407–420.

Saridis, G. N. and Valavanis, K. P., (1988). "Analytical Design of Intelligent Machines", *Automatica the IFAC Journal.*

Shannon, C., Weaver, W., (1963). *The Mathematical Theory of Communications,* Illini Books.

Stephanou, H. E., (1986). "Knowledge Based Control Systems", *IEEE Workshop on Intelligent Control 1985*, p. 116, RPI, Troy, New York.

Valavanis, K. P., (1986). "A Mathematical Formulation for the Analytical Design of Intelligent Machines", Ph.D. Thesis, Technical Report RAL No. 85, Dept. of ECSE, Rensselaer Polytechnic Institute, Troy, New York.

Vamos, T., (1986). "Metalanguages – Conceptual Models. Bridge Between Machine and Human Intelligence", Working paper E/37, Hungarian Academy of Science.

Wang, F. and Saridis, G.N., (1988). "A Model for Coordination of Intelligent Machines Using Petri Nets", Symposium on Intelligent Control, Washington, D.C., August.

Winston, P. (1977). *Artificial Intelligence*, Addision–Wesley, New York.

Zames, G., (1979). "On the Metric Complexity of Causal Linear Systems, ϵ–entropy and ϵ-dimension for Continuous Time". *IEEE Trans. Automat. Control*, Vol. AC–124, pp. 222–230, April.

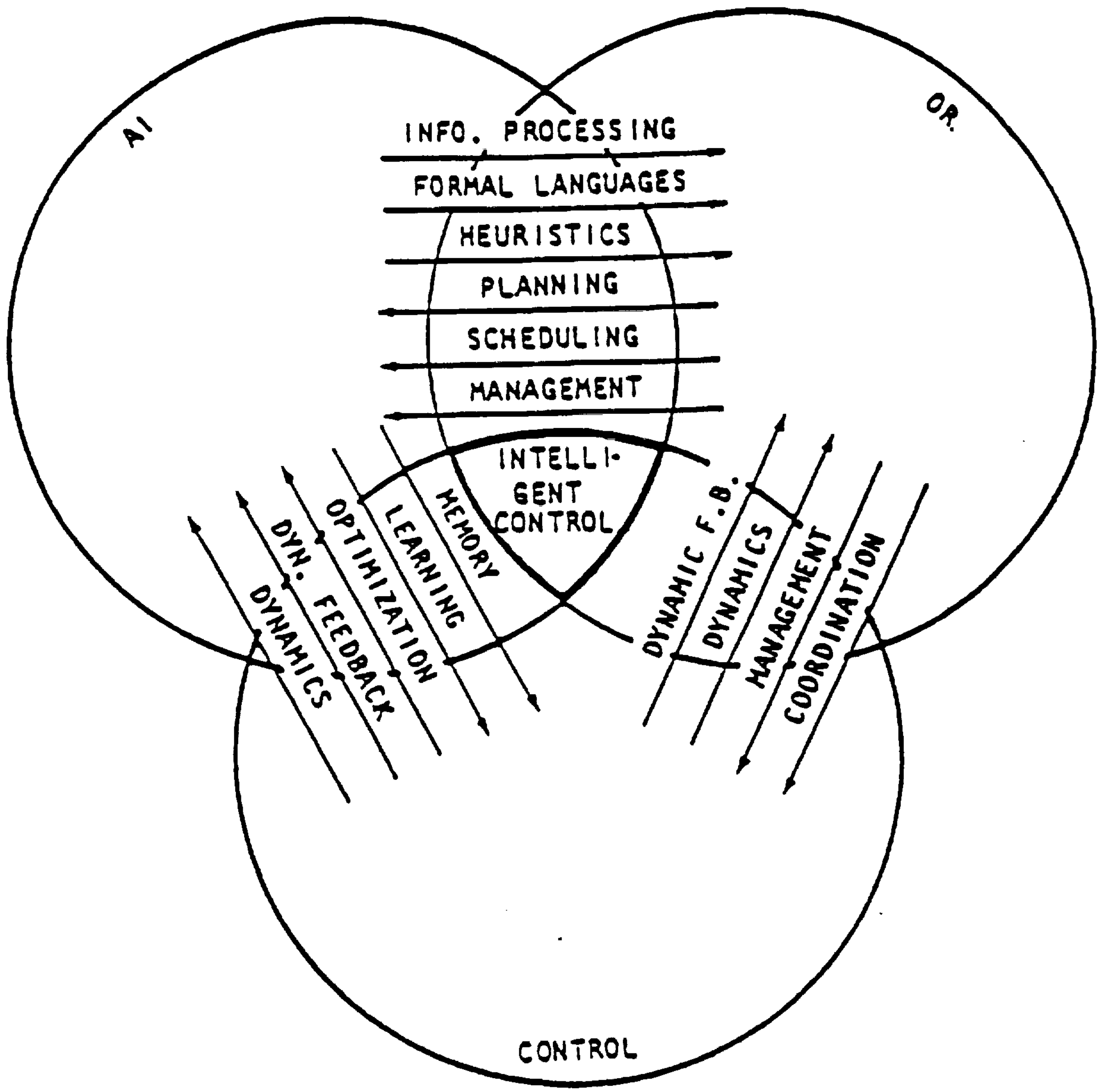

Fig. 1. Intersection of Artificial Intelligence Operations Research, and Control Theory and the Resulting Intelligent Control.

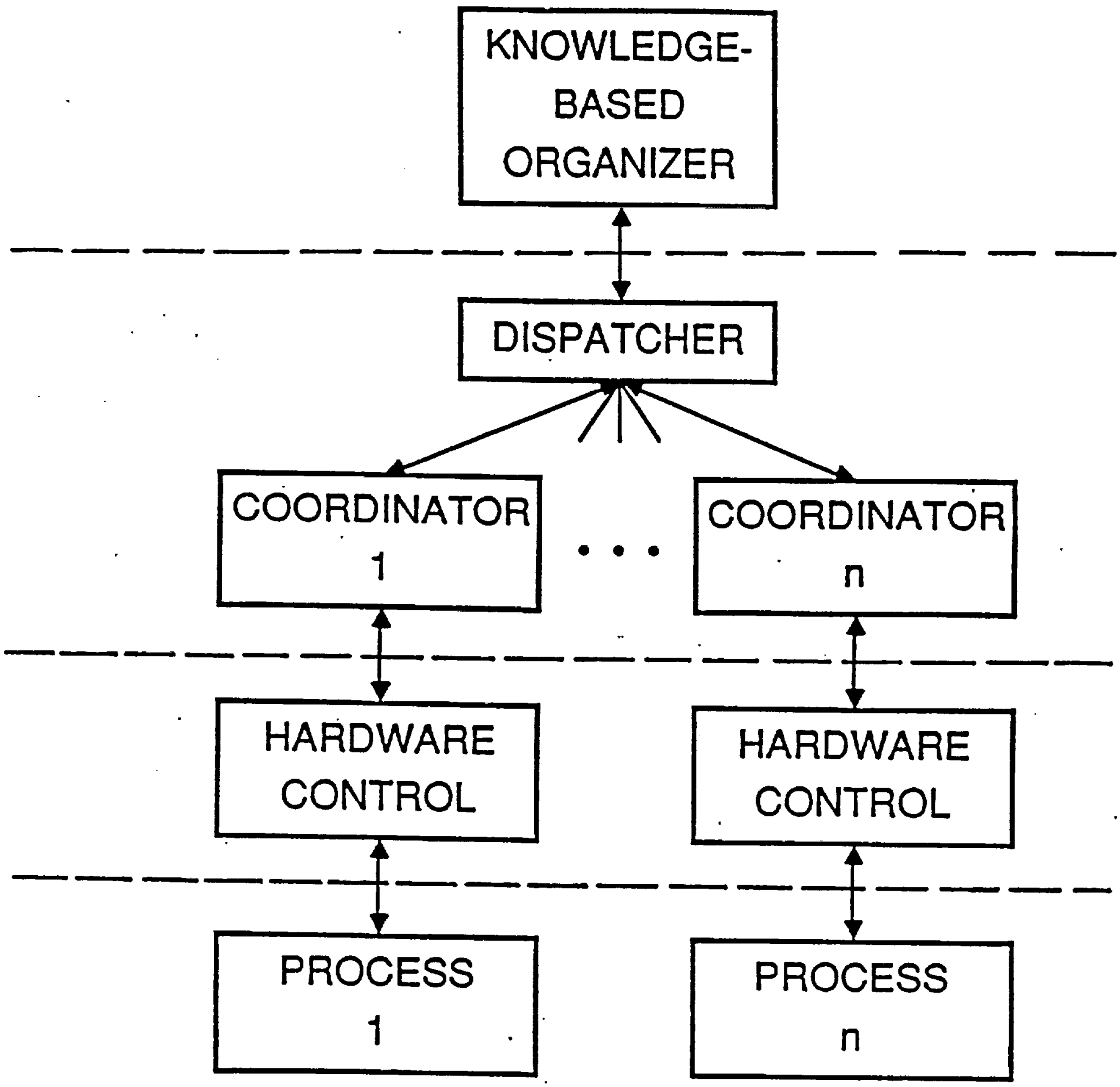

Fig. 2. Hierarchical Intelligent Control System.

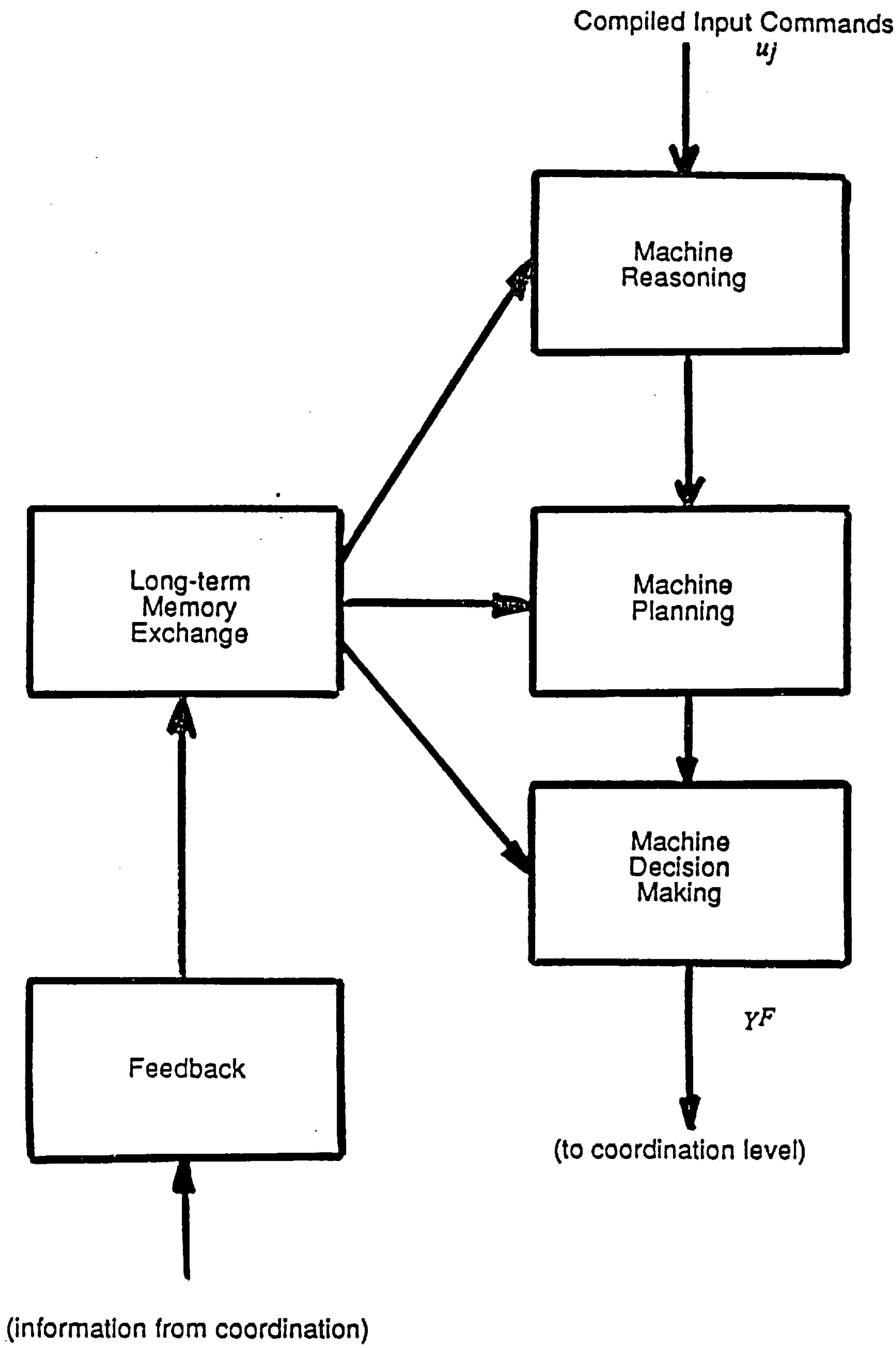

Figure 3. Block Diagram of the Organization Level

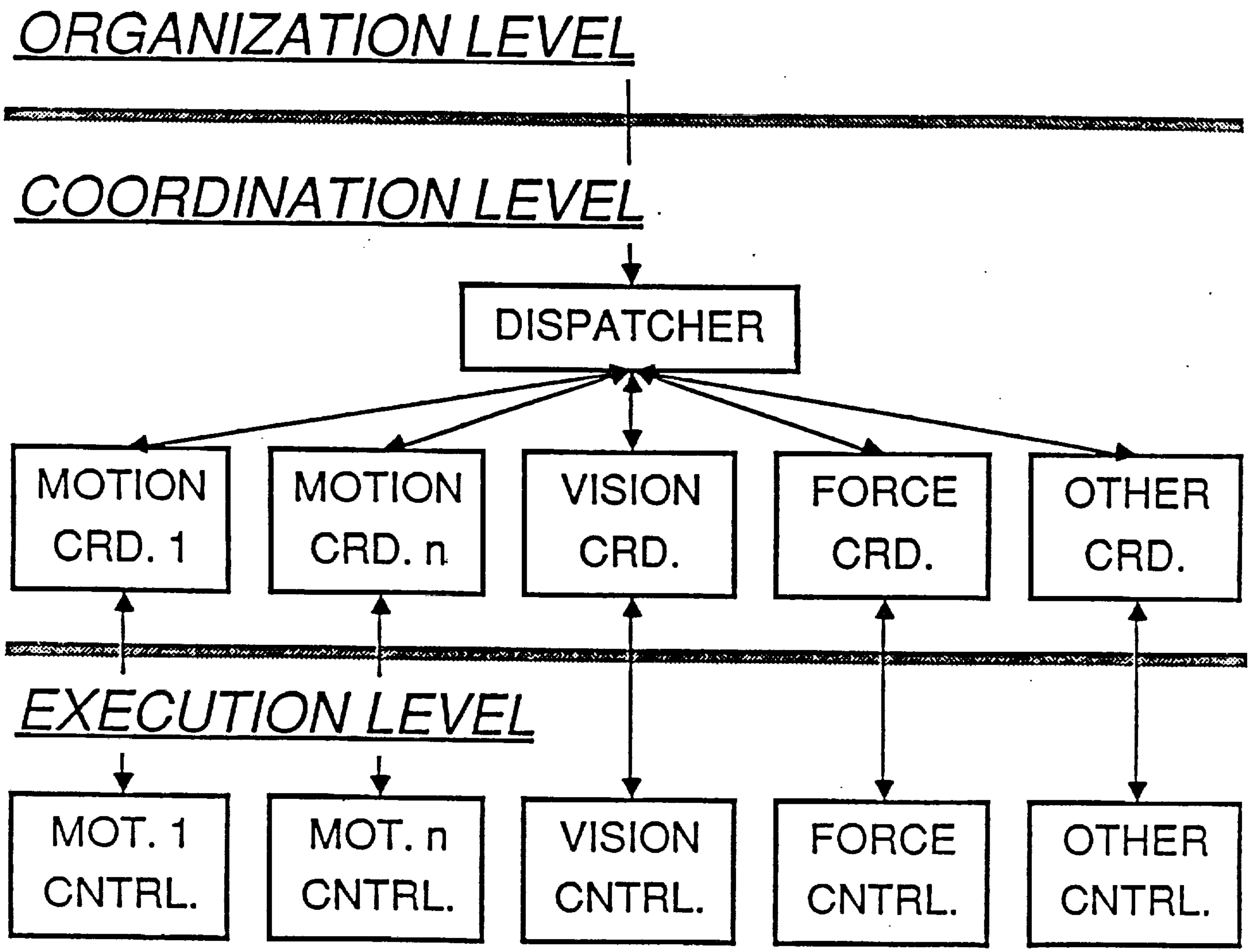

Fig. 4. Hierarchical Intelligent Manipulator Control System.

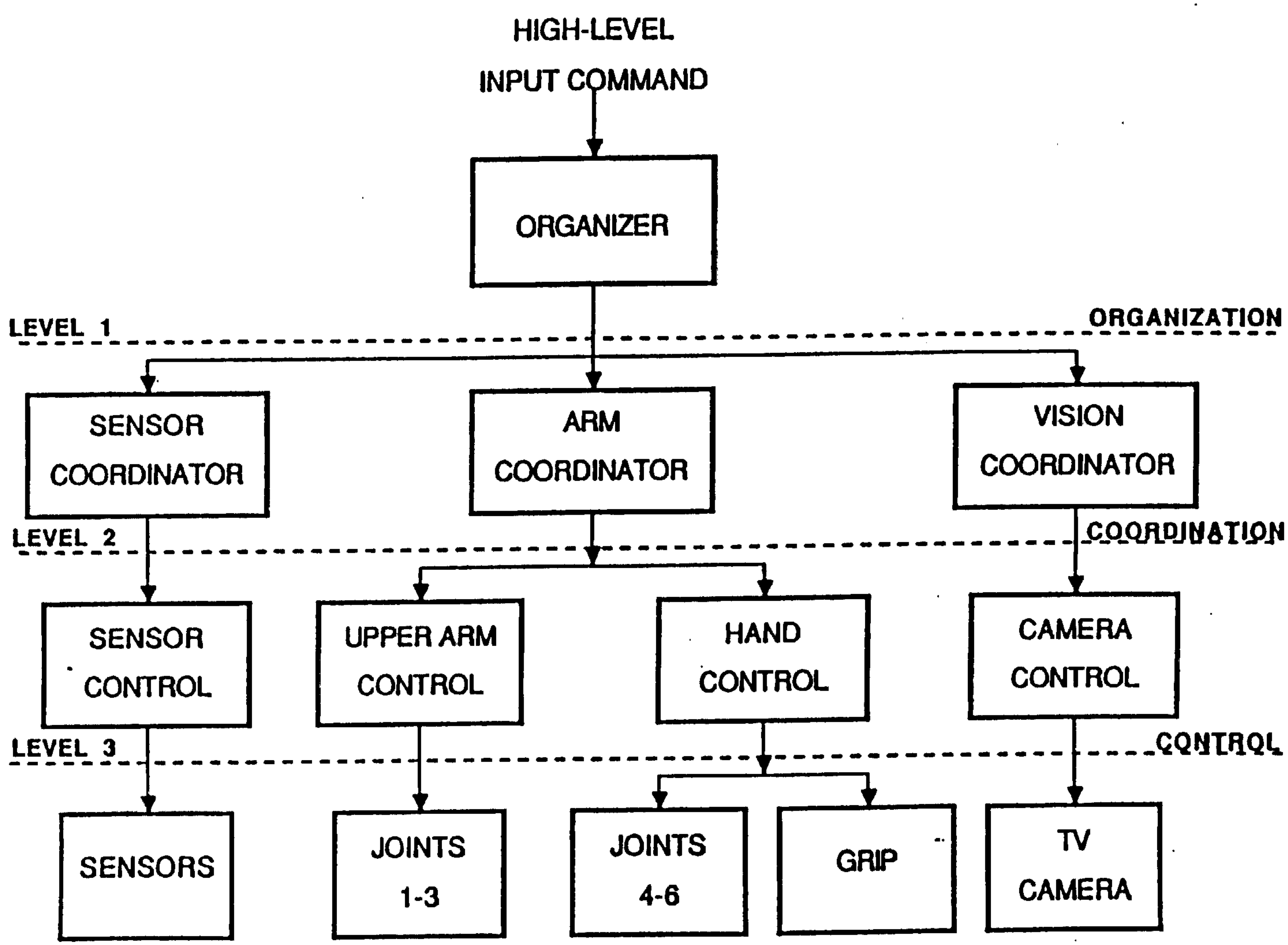

Fig. 5. Hierarchically Intelligent Control for a Manipulator with Sensory Feedback.